submitted to *Geophys. J. Int.*

Analysis of Ground Motion Simulation Validation Metrics through Semi-supervised and Supervised Learning

Naeem Khoshnevis,[1] Ricardo Taborda[1,2⋆]

[1] *Center for Earthquake Research and Information, University of Memphis, Memphis, TN* 38152*, USA*

[2] *Department of Civil Engineering, University of Memphis, Memphis, TN* 38152*, USA*

SUMMARY

Abstract here.

Key words: Time-series analysis; Numerical solutions; Numerical approximations and analysis; Earthquake ground motions; Computational seismology; Wave propagation.

1 INTRODUCTION

The validation of ground motion synthetics has received increased attention over the last few years due to advances in physics based deterministic and hybrid simulation methods. Unless in very low frequencies ($f \leq 0.5$ Hz), due to limitations and uncertainties in the ground motion simulation and velocity models, it is not possible to match the simulation and synthetics wiggle by wiggle. Therefor, in order to compare synthetic seismogram with recorded data as a complex time series, different metrics are defined. These metrics are means to characterize the how well the synthetic matches the statistical characteristic of observed data. In general, these metrics compute the goodness-of-fit (GOF) between data and synthetic in time and

⋆ Corresponding author: rtbrdros@memphis.edu

frequency domains. Anderson (2004) proposed 10 metrics to be used to characterized the GOF in different frequency bands. Kristeková et al. (2006, 2009) developed envelope and phase misfit criteria as well as GOF metrics for realistic scenarios. Olsen & Mayhew (2010) presented an alternative GOF metrics for broadband synthetics with relatively high resolution of the small misfits, and Taborda & Bielak (2013) added modification to Anderson (2004) metrics.

Recently, due to advances in computational resources, high frequency 3D ground motion simulation on a regional scale were conducted in different regions using different methods (give some references.) The accuracy of simulation with observed data are measured using these GOF scores. However, researchers use these metrics in different perspectives which is highly dependent on their application. For example, for structural engineers who are interested in designing tall buildings, among other metrics, displacement could be an important factor. However, for rigid and semi-rigid structures like dams and nuclear power plants, acceleration could be more important. In general, for all purposes, the duration of strong ground morion could be an important factor. Analysis of the strong ground motion simulation for a specific application based on individual metrics is an acceptable practice, however, validating the simulation process only with individual score is not a valid approach. Our previous studies show that metrics could have completely different scores for a pair of synthetic and data (e.g., Taborda et al. 2016). Taborda & Bielak (2013) defined a metric from a combination of Anderson (2004) scores representing a general overview of GOF. However, the metric is a linear combination and somehow is biased by the authors preferences about the metrics.

Physics-based ground motion simulation community push the maximum frequency of the simulation to higher values ($f_{\max} \geq 5\ Hz$) and there is a high demand for a metric to uniformly validate the simulations (refer to the high-f project). Although Anderson (2004), defined different level for classifying the stations as poor, fair, good, and excellent based on individual metrics, the question becomes what if there is a station with completely different scores for different metrics. This research question motivated us to approach the problem from different perspective. In other words, if we assume we know the accuracy of the simulation process (let's say we classified it as good) what is the relationship between different GOF scores for that pair of stations. However, as far as the authors knowledge, there is no metric to be able to classify a pair of stations based on GOF scores as an estimation of the overall accuracy of the simulation. Therefore, as a first step, if we could label the pair of station for simulation accuracy we can discuss the relationships between GOF scores. To do this we need a good amount of dataset where calculated the GOF scores for a good amount of stations.

Taborda & Bielak (2014) conducted a comprehensive study to analyze the functionality of different velocity models in the Los Angeles basin. They used 2008 Chino Hills earthquakes based on 3 velocity models and conducted simulations with maximum frequency of $f_{\max} = 4$ Hz. They compared the accuracy of the velocity models through using the GOF scores. The study generated an invaluable synthetic and data GOF scores dataset.

The first and the most common method to categorizing multidimensional unlabeled data is clustering and specifically the distance based methods like *k-means* (MacQueen et al. 1967). However, there is a drawback in this method. If the data is not explicitly distinguishable the final clusters are highly dependent on initial starting points of the algorithm. Therefor, for the use of stations with 11 dimensions, the ordinary *k-means* approach results in fairly different clusters after each analysis. The main reason for this issue is the algorithm is solely looking for data which are close together in terms of Euclidian distance. However, in our case we have one more extra information. Hypothetically, if we have a pair of synthetic and data with GOF score equal 3 for all metrics, according to Anderson (2004) we can classify the station as poor with a strong confidence. This is also true for GOF score of 5,7, and 9 for fair, good, and excellent classes, respectively. This background knowledge about the database could give a strong formation to the clustering process. Wagstaff et al. (2001) defined a *k-means* clustering using background knowledge. Adding this feature to the algorithm improves the functionality and most importantly converge to the same clusters with any random initial cluster's centers. Using this approach, first we cluster stations then we statistically analyze each cluster data, based on these results we remove features with different trend from the analysis. Later on, in a supervised learning algorithm, we develop a classification algorithm based on their GOF scores.

In the following section,we present a summary of the validation metrics then we discuss the dataset. Later on we explain the methodology. We provide a brief summary of unsupervised learning which is ordinary and constrained clustering approaches. We also discuss the basics of decision tree algorithm as a supervised learning method. We discuss the clustering challenges and use subspace analysis. We develop two decision tree classifiers based on labeled data for future use in comparing data and synthetics. We conclude the study by applying the generated classification methods on Taborda & Bielak (2014) dataset.

2 VALIDATION METRICS

Accuracy of simulations is evaluated based on a quantitative validation of the simulation ground motions. The validation process consist of comparisons between synthetics and data,

at locations where records were available for the simulated events. For validating the simulation of strong strong motion complex time series, different metrics are defined. These metrics evaluate the similarity of two signals in time or frequency domain.

Anderson (2004) proposed 10 individual quality measures to evaluate the credibility of synthetic seismograms. He proposed to apply the metrics on 10 frequency bands with logarithmic spacing. The logarithmic spacing puts more emphasis on lower frequencies where these frequencies are more amenable to waveform fitting and are particularly important for response of large structures. A comprehensive score (S1) could be achieved by the average of the scores of seismograms filtered in each of the valid frequency bands individually, as well as the broadband seismogram. An alternative score (S2) is obtained by averaging the scores on the ten individual criteria for only the accelerogram filtered to allow all frequencies to pass. Anderson (2004) scaled the scores into range of 0 to 10 with 10 giving perfect agreement. He suggested to consider a score below 4 as a poor fit, a score of 4-6 as a fair fit, a score of 6 to 8 as a good fit, and a score over 8 is an excellent fit. However, some of these metrics like integral measures are potentially very easy to fit and some of them like cross correlation is by far the most difficult of all parameters to achieve high score. Therefore, for a pair of synthetic and data some of metrics could belong to good and some of them could belong to fair fit classes.

In other study, Kristeková et al. (2006) developed quantitative misfit criteria for comparison of seismograms. The misfit criteria are defined based on the time-frequency (TF) representation of the seismograms obtained at the continues wavelet transform. Based on local TF envelop and phase differences, they defined envelop and phase misfit dependent on both time and frequency. With projection of TF misfit onto frequency or time domain, they computed frequency- or time-dependent misfits, respectively. The method proposed by Kristeková et al. (2006) needs to consider one of signals as a reference signal. Therefore, Kristeková et al. (2009) presented an extension of the theory of the TF misfit criteria. They defined locally and globally normalized TF criteria where locally normalized misfits can be used if it is important to investigate relatively small part of the signal. The globally normalized misfits can be used for quantifying an overall level of disagreement. In the extended version of the metrics, they defined a misfit criteria for 3 components signals and also they developed TF GOF criteria for more realistic scenario where there is a high level of disagreement and it is not possible to consider one of signals as a reference signal. They classified the GOF scores as poor, fair, good, and excellent based on GOF numerical values.

Olsen & Mayhew (2010) argued that the Kristeková et al. (2006, 2009) metrics are suitable for low frequency signals. Therefor, they proposed a new GOF method for validation of Broadband synthetics. They used some of metrics that are used by Anderson (2004), however, they used different scoring approach which generates a relatively high-resolution representation of the small misfits. They also used inelastic and elastic response spectra ration (IE ratio) as a structural engineering specific metric. They proposed to combine the metrics for final decisions based on user defined application-based weights.

Recently, Taborda & Bielak (2013) conducted a simulation for 2008 Chino Hills earthquake and proposed a minor modification in the Anderson (2004) metrics. They added an eleventh parameter to the Anderson's scores as strong motion duration.

A comprehensive validation approach should include metrics in both time and frequency domain. All mentioned metrics (i.e., Anderson 2004; Kristeková et al. 2006, 2009; Olsen & Mayhew 2010) have this characteristic. However, in this study we are mostly dependent on available data based on Taborda & Bielak (2014) where they used the GOF method proposed by Anderson (2004), with minor modification introduced by Taborda & Bielak (2013). The method compares synthetics against data using 11 individual parameters, namely: Arias Intensity Integral (C1), Energy Integral (C2), Arias Intensity Value (C3), Total Energy (C4), Peak Acceleration (C5), Peak Velocity (C6), Peak Displacement (C7), Response Spectra (C8), Fourier Amplitude Spectrum (C9), Cross Correlation (C10), and Strong Phase Duration (C11). Each parameter is mapped on to a numerical scale ranging from 0 to 10, where a score of 10 corresponds to a perfect match between two signals.
Following the guidelines suggested by Anderson (2004), the scoring procedure is applied to each pair of data and synthetic using compatible broadband sets, and series of bandpass filtered versions of the signals or sub bands, SB1, SB2, SB3, SB4, and SB5. Taborda & Bielak (2014) represented the signals bandpass filtered between 0.1 and 4 Hz for BB, and between 0.1 Hz and 0.25 Hz, and 0.25 and 0.5 Hz, and 0.5 and 1 Hz, and 1 and 2 Hz, and 2 and 4 Hz for SB1, SB2, SB3, SB4, and SB5, respectively. They also reported a scaled combination score of metrics and bands. However, since the combined scores are linear scaled combinations of metrics and frequency bands, in this specific study, they do not add new information to the data base. Khoshnevis & Taborda (2015) showed that filtering approach could affect the GOF score and change the final classes. Therefore, in order to avoid extra source of uncertainty in

the study, we only consider broadband GOF scores ($0.1 - 4Hz$). Analysis of effect of bands in the results could be an appropriate subject for another study.

3 GOODNESS-OF-FIT DATABASE

State of California is an earthquake prone region with a long history of seismicity. The Los Aneles basin, due to its unique geologic sediments, has been studied from different perspective. Numerous high resolution velocity models are developed mainly to conduct forward ground motion simulation. Occurrence of frequent moderate magnitude earthquakes made the Los Angeles basin as a natural laboratory for seismic studies specially testing the forward ground motion simulation modeling (give some references including Tabarda et al 2016). In a relatively recent study, Taborda & Bielak (2014) conducted a comprehensive study in order to assess the functionality of developed velocity models in the Los Angeles basin and surrounding areas. They simulated the 2008 Chino Hills earthquake using 3 velocity models including: CVM-S4, CVM-H, and CVM-H+GTL (for more details about these models see Taborda & Bielak (2014).) They computed the GOF scores based on Anderson (2004), with minor modification introduced by Taborda & Bielak (2013) as discussed in the validation section.Fig. 1 shows the location of the epicenter and stations' locations.

We use Taborda & Bielak (2014) database where they computed 11 scores for 3 different velocity models. In this study we only analyze the results for broadband which in this case is $f = 0.1 - 4$ Hz. Fig. 2 shows the box plot of all data that we used in this study which we separate it in velocity models and components.

As we explained before, we use all data set regardless of velocity model and components. However, distinguishing data for velocity model and components could answer the question that if the goodness of fit scores dependent on them. Addressing this research question is beyond the scope of this paper. We distinguish data based on velocity models and components only in presenting data or results.

Fig. 3 shows the density distribution of each score for CVM-S4.

4 METHODOLOGY

As we discus in section.2, there is no consensus on a single metric who able to estimate the accuracy of the simulation or could classify the simulation based on given GOF scores. On the other hand we have a comprehensive dataset. Therefore we need to conduct a data mining and knowledge discovery process to get the classification pattern out of the data set. A

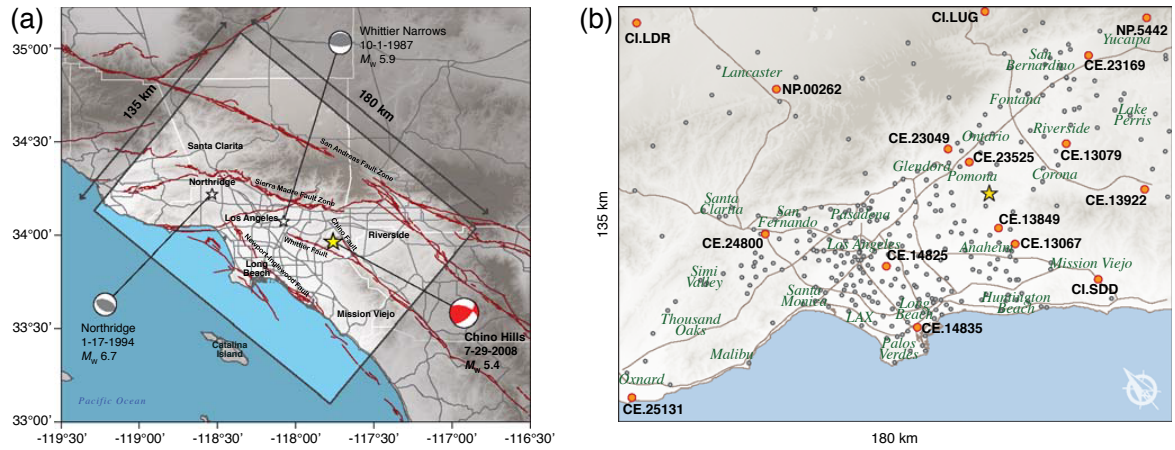

Figure 1. (a) Region of interest and epicentral location of the 2008 Chino Hills earthquake (highlighted). Focal mechanisms are shown at the margins of the region of interest. Major quaternary faults in the area are shown in the back along with the main roads and county divisions. (b) Simulation area of interest. Dots indicate the location of the 336 stations considered in this study for validation. A subset of these stations include labels and highlights with larger symbols for later reference. The main local and interstate roads are also shown here. In both panels, the background shows the hillshade topography of the region. The color version of this figure is available only in the electronic edition. (figure and caption from 2014 paper. needs revision)

common method for generating a decision rule having disjunctive characteristic is decision tree approach. The method is developing a rule for classifying data (here stations) based on different attributes (here GOF scores), this process also called supervised learning in machine learning community. However, decision tree needs labeled data. In our case labeled data would be a data with another attribute which shows the overall accuracy of the simulation (In other word the one metric that we are developing methods to generate it). Therefore we need to go one step back and generate label for the data. Generating labels for data is categorized as unsupervised learning or clustering. Different methods have been developed for clustering. The most commonly used method is *k-means*. However, *k-means* method gives different results based on initial values of cluster center especially where data is not clearly distinguishable. In result we use modified *k-means* approach. This method use background knowledge in the unsupervised learning process. Using background knowledge in unsupervised learning converts it into semi-supervised learning process.

In summary, first we label the data based on the constrained *k-means* approach, then we statistically analyze the each group results. We expect to see similarity among attributes within each classes. Later on, in a classification process, using decision tree we find the best hypoth-

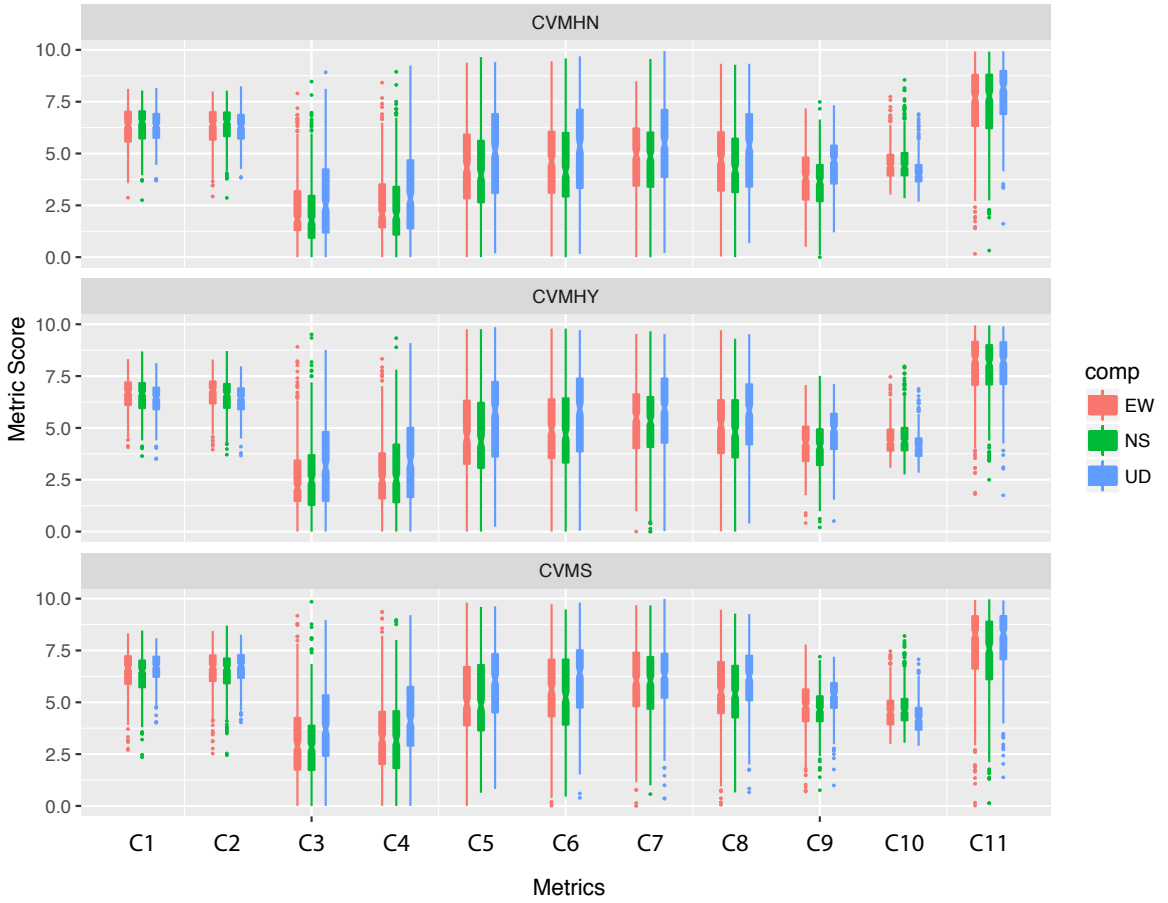

Figure 2. Box plot of data used in the study. Metrics are shown for C1 to C11 for 3 components and in separate plots for velocity models. Median of data are shown as a notch. Thick lines represent the IQR (Interquartile Range, Q3-Q1) of data. Outliers (data less than Q1-1.5*IQR and greater than Q3+1.5*IQR) are shown as scatter dots above and below plots if applicable.

esis to classify a station based on GOF scores. In the rest of this section first we discuss the clustering method for both ordinary and constrained *k-means* approach then we discuss the decision tree algorithm and basics in separating data. We discuss the statistical analysis of the results in the results section.

4.1 Clustring

Clustering is an unsupervised approach for grouping of data based on measure of similarity and it is considered as an exploratory activity as a part of data mining process (Fayyad 1996). In each valid cluster, patterns are more similar in each other than they are to pattern belonging to a different cluster. Many clustering algorithms are developed for different ap-

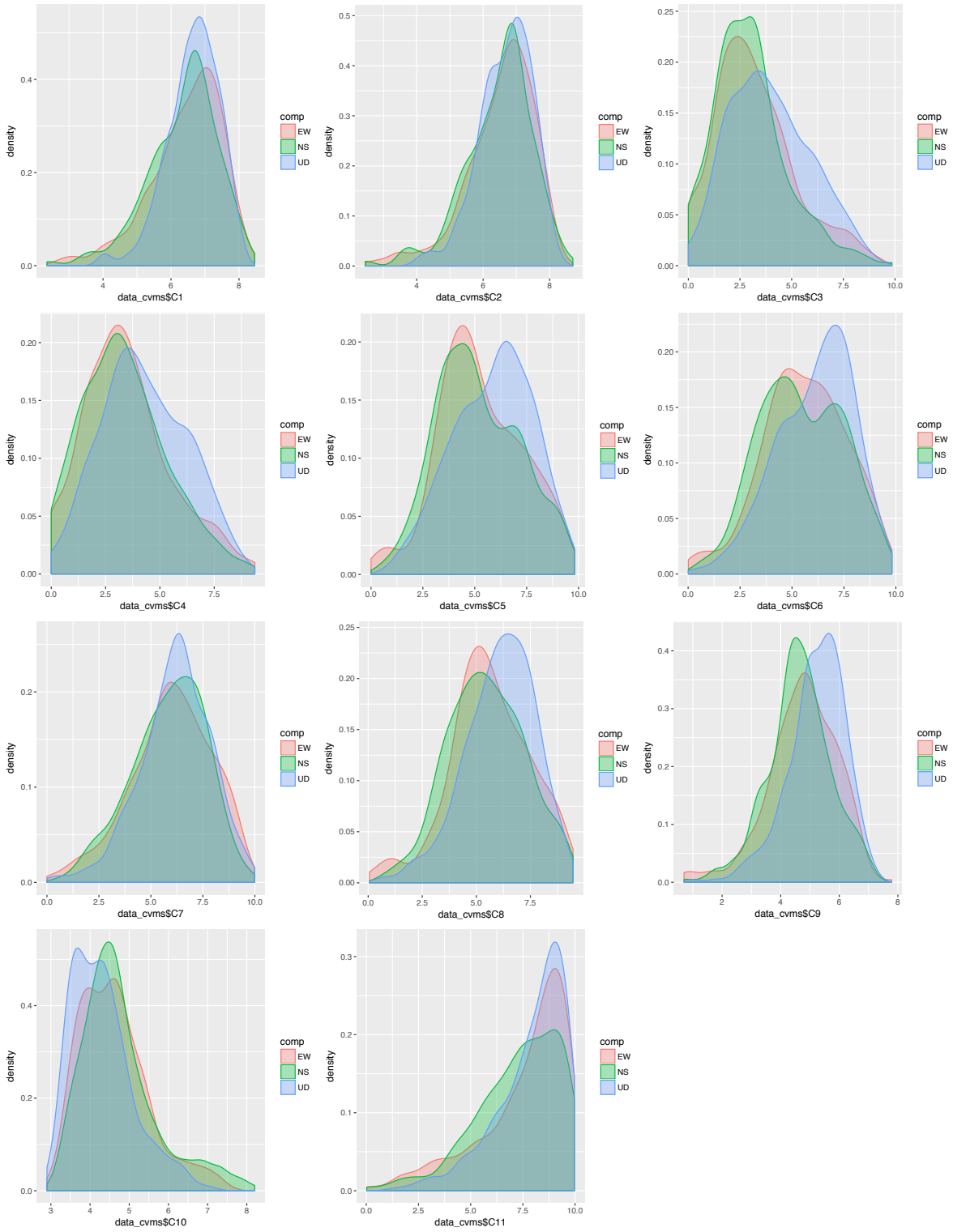

Figure 3. Distribution of data (only CVMS-4) for different components in different scores based on different components. Each distribution has a unit area.

plication and study the difference of them is beyond the scope of this paper. In general, at the top level, Jain et al. (1999) distinguished the clustering approach into Hierarchical and Partitional approaches. Aside from differences in application and technical details in implementation, hierarchical methods produce nested series of partitions, while partitional methods produce only one. In this study we are interested in using partitional, distance based clustering algorithm, which is also know as *k-means* algorithm. MacQueen et al. (1967) described a process for partitioning an *n-dimensional* population into k sets on the basis of a sample. The process appears to give partitions which are reasonably efficient in the sense of within-class variance. For numerical values, *k-means* algorithm starts with a user defined number of clusters (k) and assign a random mean value for each cluster (or randomly choose k data and assign them as cluster centers.) Then it computes the distance of data and cluster centers. A variety of distance measures are in use in different studies, however, we use the Euclidian distance through

$$d(x_i, x_j) = \sqrt{\Sigma_{k=1}^{n}(x_{i,k} - x_{j,k})^2}, \tag{1}$$

where n is the number of features and d is the distance of x_i and x_j in *n-dimensional* domain. After computing the distance of the points from each clusters' mean (center), the algorithm continues with labeling the data after the closest cluster. At the next iteration, it computes the mean value of data for each cluster and updates the clusters' centers. The algorithm repeats the steps unless the amount of updates among the cluster centers is less than a tolerance value. *k-means* clustering has been applied in different applications, however, there is no clustering technique that is universally applicable in uncovering the variety of structures present in multidimensional datasets. Therefore, clustering is a subjective process. The same set of data items often needs to be partitioned differently for different application. In result, it is essential for the user of a clustering algorithm to not only have a thorough understanding of the particular technique being utilized, but also to know the details of the data gathering process and to have some domain expertise; the more information the user has about the data at hand, the more likely the algorithm would be able to succeed in assessing its true class structure. Domain concept can play several roles in the clustering process, and a variety of choices are available to the practitioner. For example, one can add an additional feature to the database (Jain et al. 1999).

The major problem with *k-means* algorithm is that it is sensitive to the selection of the initial partition and may converge to a local minimum of the criterion function value if the initial partition is not properly chosen. Another problem accompanying the use of *k-means*

algorithm is the choice of the number of desired output clusters. Several variant of the *k-means* algorithm have been reported in the literature. Some of them attempt to select a good initial partition so that the algorithm is more likely to find the global minimum value, another variation is to permit splitting and merging of the resulting clusters (Jain et al. 1999).

In our application the number of clusters is not a problem and there is a consensus among researchers in the number of clusters (i.e., poor, fair, good, excellent), however, our initial attempts represent that the results are highly sensitive to the initial clusters' center.

Every clustering algorithm uses some type of knowledge either implicitly or explicitly. In this study the background knowledge is a series of hypothetical stations. We assume that there are four stations with score of 3, 5, 7, and 9 for all of their metrics. Based on the score limits in section. 2 we know these stations belong to poor, fair, good, and excellent classes, respectively. These background knowledge could help the clustering processing to be in right direction regarding the fact that increasing dimension of data could increase noise in clustering and cause difficulty to better partitioning. Wagstaff et al. (2001) demonstrated a modification of *k-means* clustering algorithm which uses the background information of the domain or dataset. The algorithm adds two types of constraints to the clustering including:

- Must-link: constraints specify that two instances have to be in the same cluster.
- Cannot-link: constraints specify that two instance must not be placed in the same cluster.

The algorithm is described in detail in Wagstaff et al. (2001), however, in simple words, in the ordinary *k-means* process before assigning data to the closes cluster, it controls the must-link and cannot-link conditions. Therefor, in this condition, the closest mean of cluster is not necessarily the final cluster of the data. Fig. 4 represents the difference between ordinary and constrained k-means clustering approach using 4 points and 4 cluster centers. Fig. 4.a represent the *k-means* without constrained. According to the definition and the explanation in this section, the closest cluster center for each points will get the points. Fig. 4.b represents the constrained *k-means* approach where, the closest cluster center is not necessarily the final configuration. We distribute data among clusters (in this case to satisfy the Cannot-link criteria) where as the distance between cluster centers and points to be minimum. The visual inspection of the figures also confirms the accuracy of the method.

The end product of the clustering process is groups of data. Analysis of these groups, individually, gives the idea about the clusters in terms of within class variation. In these analysis we mainly study the behavior of different features in each cluster and isolate only the

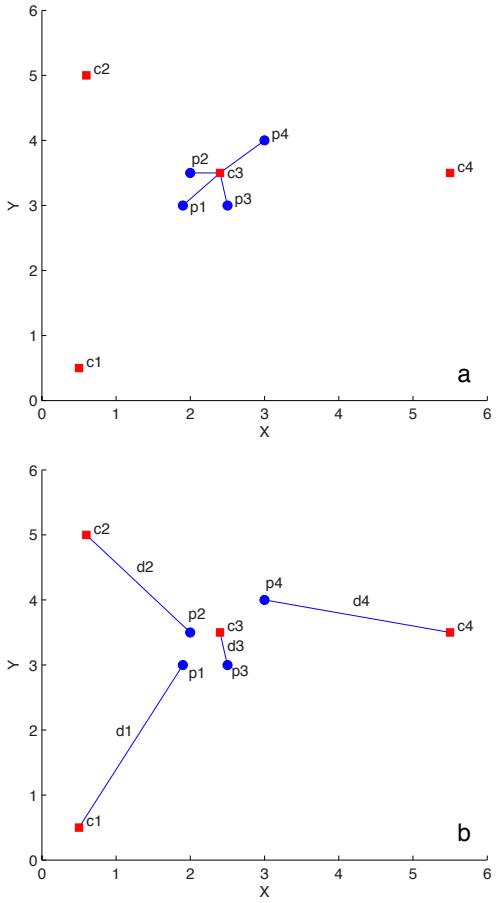

Figure 4. Representation of ordinary (a) and constrained (b) *k-means* approach. There are four data points (p) and four cluster centers (c) as a sample of 2-*Dimensional* dataset. All points are defined as cannot-link constraints. In other words, the points cannot be in the same cluster. In the constrained *k-means* approach we redistribute the points such that the $\sum d$ becomes minimum.

most descriptive features to be used in the supervised classifier that assumes a given number of classes in the data set. In the next section we provide a basics of decision tree algorithm.

4.2 Decision Tree

Decision tree learning is a method for approximating a target function, in which learned function is represented by decision tree (Mitchell 1997). Quinlan (1986) demonstrated that the technology for building decision trees from examples is fairly robust. He summarized an approach to synthesizing decision trees that has been used in a variety of systems, and he describes one such system, ID3, in detail. Although ID3 algorithm is successful in most classification problems, it does not handle the numeric attributes and only one attribute at a time is tested for making decisions. Quinlan (1993) extended the ID3 algorithm and presented

the C4.5 algorithm. Later on they added boosting capability to C4.5 and called it C5.0. The improved algorithm accepts both continues and discrete features and solves over fitting problem by pruning. Although there are many other algorithm for implementing decision tree, in this study we use the C5.0 algorithm. Discussing the algorithm in details is out of the scope of this paper. There are fairly good amount of references to study those details (e.g., Mitchell 1997; Quinlan 1993; Hornik et al. 2009).

In general, decision tree measures the effectiveness of an attribute in classifying the training data through information gain. According to Mitchell (1997), information gain of an attribute is the expected reduction in entropy caused by partitioning the examples according to this attribute. If the target can take on c different values the entropy is defined as

$$Entropy(S) = \sum_{i=1}^{c} -p_i \; log_2 \; p_i, \tag{2}$$

where p_i is proportion of S belonging to class i. Entropy measures the homogeneity of examples or dataset. Higher entropy means data is fairly uniformly distributed among different classes. On the other hand lower entropy means data unequally distributed among classes. The extreme case happens when all data belong to one class which results in entropy equal to zero. Given entropy as a measure of impurity in a collection of training example the measure of the effectiveness of attribute is defined through information gain which is expected reduction in entropy caused by partitioning the examples according to this attribute. Gain of the attribute A is defined as

$$Gain(S, A) \; = \; Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) \tag{3}$$

where $Values(A)$ is the set of all possible values of attribute A, and S_v is the subset of S for which attribute A has value v. Information gain is used by the algorithm in order to determine which is the better attribute for classifying the training example. The attribute with higher information gain will be the first option to separate the training data based on and grow to the lower nodes (For more detailed examples refer to chapter 3 of Mitchell (1997)).

We discussed the basic idea behind the decision tree algorithm, however, there are more details specially in dealing with outliers and noises which we address them through pruning approaches. In this study we use C5.0 package (Kuhn et al. 2015) in R to conduct the decision tree algorithm. We discuss the input parameters in the section.6.

5 CLUSTERING ANALYSIS

We start with clustering all dataset with 11 features. Visual presentation of results above 3 dimension is not possible. Therefor we present the clustering results based on a pair of features in Fig. 5. The figure represent a part of results in clustering with 11 features. There are 55 other possibility to present (e.g., C1 vs C7 or C2 vs C9). As one can see in some combination of features the clustering patterns are easily seen. For example in C3 vs C4 we can distinguish 4 region for clusters, although they are partly mixed together. This also correct for C6 vs C7 and C7 vs C8. In the latter one we observe less mixture of data than the former one. Also there are other types of figures where one feature does not add so much information to the clustering process. For example in C9 vs C10, C10 could be considered as irrelevant feature. Irrelevant feature does not add further information for clustering and redundant feature add the equivalent information as other feature (Dy & Brodley 2004). The last combination of features are a good mixture of data and it is not possible to see the clustering boundaries. C1 vs C2 and C10 vs C11 are examples of such feature combination. It is worth to mention that at this point the cluster numbers are not consistent with ordinal categorical numbers of 1,2,3, and 4 representing poor, fair, good, and excellent, respectively.

Although we don't expect to see the pattern that resulted from clustering using 11 features in presentation of only two, however, different studies show that the higher dimension reduce the effect of similarity based on distance. Parsons et al. (2004) presented an illustrative example to show the effect of dimension in reducing the importance of the distance. Effect of higher dimensions in clustering has been well studied and many different methods are proposed to reduce this effect. Among them we can name feature selection before, during, and after clustering, hybrid methods that use combination of methods to select the best subset of feature and subspace analysis. Dy & Brodley (2004) addressed two issues involved in developing an automated feature subset selection algorithm for unlabeled data. They illustrated the irrelevant and redundant features and proposed methods for evaluating candidate features using two performance criteria.

Subspace analysis is another technique to address the challenges with higher dimensional data in clustering process. Subspace clustering is an extension of traditional clustering which looks for different pattern using subset of features. Parsons et al. (2004) provided a list of algorithm for conducting subspace clustering and also some potential applications for them. The most common factor among these algorithm is the process to find the a group of best subspaces through optimization process. An n-dimensional dataset has 2^n subspaces where it could be very costly and time consuming to evaluate all of them.

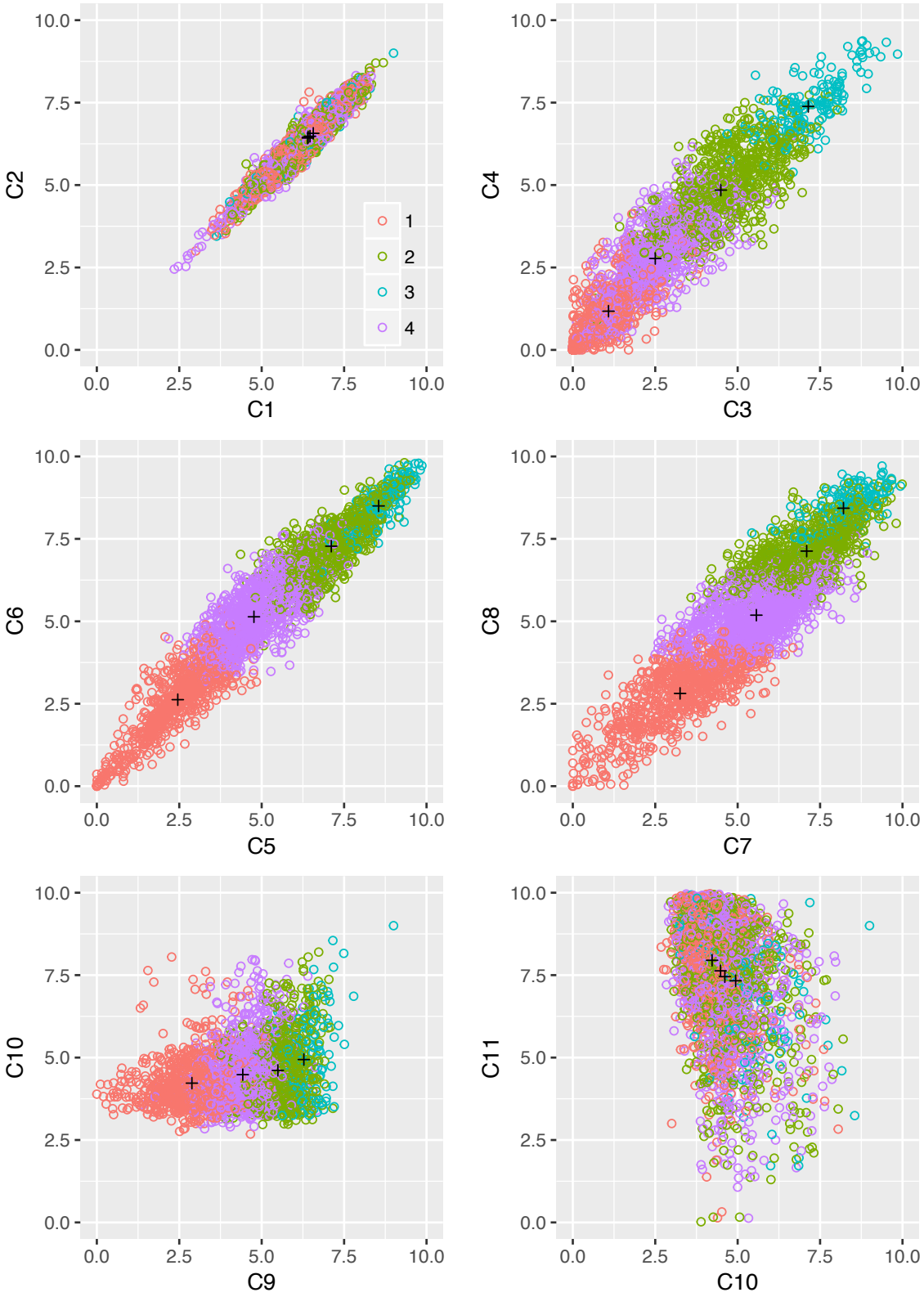

Figure 5. Results of clustering analysis with all 11 attributes (metrics). This results are subsample of 55 possible combination of metrics. Center of cluster is shown as crosses. At this plot ordinal, categorical clusters' numbers are not necessarily represent the poor, fair, good, and excellent.

As we discussed earlier, there is not standard measures for evaluating clusters in the clustering literature also there is no single clustering assignment explains every application (Dy & Brodley 2004; Jain & Dubes 1988; Hartigan 1985). The clustering process is strongly dependent on the application. A good example of the concept is clustering a whale, an elephant, and a tuna fish (Jain et al. 1999; Watanabe 1985). Whales and elephants should be in the same cluster, because they both are mammals. However, regarding the scope of the clustering the user could put tuna fish and whales in the same cluster because both groups live in water. In this study we are interested in using those features who, in general, represent the simulation accuracy in 4 different categories. As a first step we used the *k-means* clustering approach for all features, however, because of mentioned reasons the results are not easy to discuss or even evaluate. Although we have four groups of data, the question is which one should be considered as poor, fair, good, or excellent groups. Therefore, we apply a modified method of subspace clustering approach to cluster the stations. As we discussed earlier and presented in figures the constrained *k-means* method effectively put the stations in a cluster with considering the fact that our constrain stations should not be in the same cluster. High number of iteration leads the clustering process to follow the clustering concept that we are looking for which is clustering stations as poor, fair, good, and excellent. In our case number of possible subspace is 2^{11} where each features have two options wether belong to subspace or not. However, because of preserving the distance based criteria effects we limited the number of features in the subspace to be 2,3, and 4 features. Therefor we have 330,165, and 55 unique subspaces for 4D,3D, and 2D, respectively. We conduct a *k-means* clustering analysis for each of these subspaces. After considering the constraint conditions, which is assigning cannot-link stations to different cluster and and repeat the *k-means* algorithm, in some cases, it is not possible to distinguish all four cannot-link stations in different clusters. In this study we ignore these cases. We only use those combination of features that gives 4 unique clusters for the constraint points (hypothetical stations), therefore, we know for sure that all data within same cluster let's say with metric value 3, should be considered as poor. We also control the clusters to be consistent (we reformat numbers to assign cluster 1 to all group of stations that our first constraint belongs to them and so on). Finally, using 550 unique subspace clustering results, we assign the most frequent class to the station.

Fig. 6 shows the results for different velocity models and components. A preliminary visually inspection shows that the behavior of scores with respect to the velocity model is similar. Therefor we merge data of different velocity models. Fig. 7 shows the results.

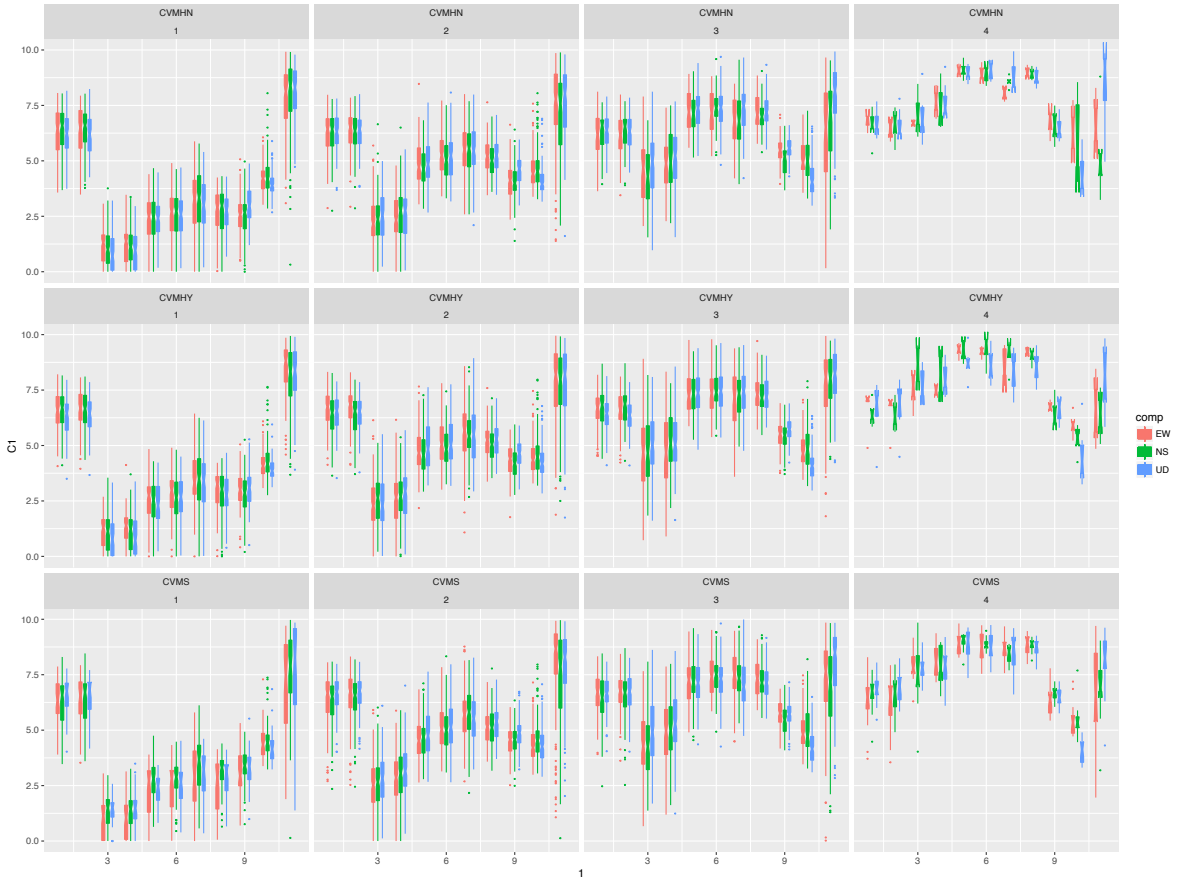

Figure 6. Box plot of data used in the study. Metrics shown for C1 to C11 for 3 components in separate plots for velocity models and clusters. Median of data are shown as a notch. Thick lines represent the IQR (Interquartile Range, Q3-Q1) of data. Outliers (data less than Q1-1.5*IQR and greater than Q3+1.5*IQR) are shown as scatter dots above and below plots if applicable.

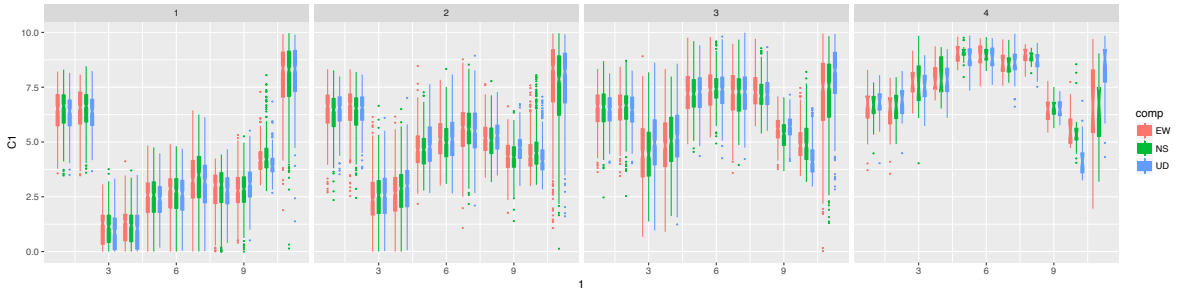

Figure 7. Box plot of data used in the study. Metrics shown for C1 to C11 for 3 components in separate plots for clusters. Median of data are shown as a notch. Thick lines represent the IQR (Interquartile Range, Q3-Q1) of data. Outliers (data less than Q1-1.5*IQR and greater than Q3+1.5*IQR) are shown as scatter dots above and below plots if applicable.

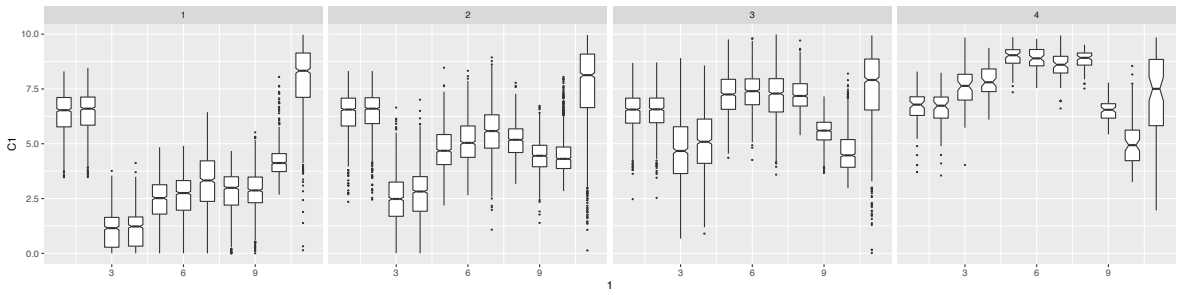

Figure 8. Box plot of data used in the study. Metrics shown for C1 to C11 in separate plots for clusters. Median of data are shown as a notch. Thick lines represent the IQR (Interquartile Range, Q3-Q1) of data. Outliers (data less than Q1-1.5*IQR and greater than Q3+1.5*IQR) are shown as scatter dots above and below plots if applicable.

Although there are some differences in the scores based on different components, there are general similarity in all clusters for different metrics among their components. Fig. 8 shows the results with merging the components.

Fig. 8 represents the results from a broad perspective. It is necessary to do statistical significance tests to understand the effect of component, velocity model, earthquake, and frequency band in the results. According to Fig. 8 scores C1, C2, and C11 has the least changes in different clusters. They suggest the idea that they may not helpful in defining a decision rule. On the other hand, other scores are fairly variable among different clusters. Scores C1, C2, and C10 are very similar in cluster 1 to 3. However, we can see a small increase in cluster 4 for these scores. Also the data variation becomes small and range of data moves higher. Scores C3-C9 have a considerable increment from cluster 1 to cluster 4. Although there are a broad variation of data in cluster 3 and cluster 4 for these scores, good amount of data (50% based on the IQR) are in a small range and increasing with respect to the clusters. C11 has a constant behavior with a small reduction in median at cluster 4. Fig. 9 shows number of stations in each cluster based on component.

6 CLASSIFICATION MODEL

In the section. 4 we discussed the basics of decision tree method and we provided a brief history of available algorithm. In this section we develop a classification model. Although there are numerous methods to classify labeled data, however, in this study we are interested in a method to not only predict the class of simulation based on metrics, but also give an intuition about the decision process. Therefor we use tree based algorithm to develop the prediction

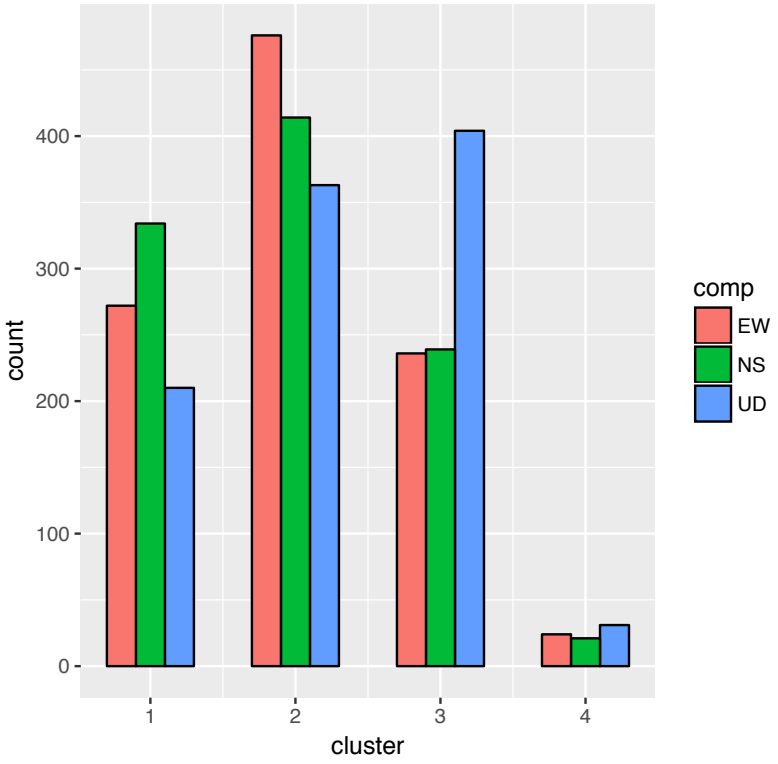

Figure 9. Number of data in each cluster based on components after conducting constraint *k-means* clustering through subspace analysis.

model.

After preprocessing data the most important step in developing a prediction model is feature selection. The step becomes more relevant when data set has hundreds to tens of thousands of variable or features. Although our dataset does not have numerous features we do some analysis on features before developing the prediction model. Features of a dataset could have similar behavior. This characteristic is easy to understand by correlation criteria. The Pearson correlation coefficient between two features is defined as

$$R(i) = \frac{cov(X_i, X_j)}{\sqrt{var(X_i)var(X_j)}}, \tag{4}$$

where *cov* is covariance function and *var* is the variance function (Guyon & Elisseeff 2003). Fig. 10 represent the Pearson Correlation factor among features.

As we can see from the figure some metrics are highly correlated. Among them we can mention C1 vs C2 , C3 vs C4, C5 vs C6 and C8, and C6 vs C8 have correlation factor more than 90%. Also we observe that C11 is mostly have a negative correlation with other features.

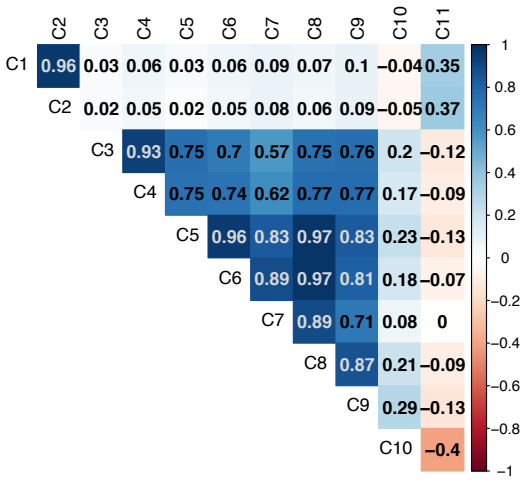

Figure 10. Pearson correlation values between metrics. The plot is generated using Corrplot package in R (Wei & Simko 2016)

Feature selection is a broad research field. Guyon & Elisseeff (2003) presented a review of common methods to select features, construct new features or reduce the domain. They highlighted the fact that redundant variable or variables that are useless by themselves may be useful in combination with other variables. Clustering is a method to construct new features as an unsupervised learning or feature selection without having the labels in calculation. The idea is to replace a group of features mean with newly developed features. It helps to reduce the number of features and use all of the available information (Duda et al. 1973; Guyon & Elisseeff 2003). Since we don't have a long list of features and also we want to study all feature effects we leave all features to be used in the prediction model.

After analysis of each features the next step is subsetting data for training and testing process. A subset of data that is used for training and test should have about the equal distribution of target values. Equally distributed data helps training process to not biased to an special class also helps to have an accurate evaluation of the model during the test process. Imbalanced data may decrease the classifier performance. A dataset is imbalanced if the classification categories are not approximately equally represented. Branco et al. (2015) provided a comprehensive review of different approaches toward imbalanced distribution of target variables. A mostly common approach is data preprocessing through resampling. According to Branco et al. (2015), resampling strategies based on diverse set of techniques such as: random under/over-sampling, distance methods, data cleaning approaches, clustering algorithm among them.

According to the clustering results, among all data with different components and velocity

Table 1. Confusion Matrix for binary problem

		Prediction	
		Positive	Negative
Reference	Positive	TP	FN
	Negative	FP	TN

model there are 816,1253,879, and 76 data belong to cluster 1,2,3, and 4, respectively. Weiss & Provost (2003) discussed the fact that the perfectly balanced dataset does not always provide the optimal results. One option could be under sampling cluster 1 to 3 to be in the similar distribution of cluster 4. This may eliminate the useful examples leading to a worse performance. Therefor we 10 times over-sample the data with cluster 4 to have a distribution similar to other cluster. Oversampling may increase the likelihood of overfitting, however, in this case the oversampling rate is not high. On the other hand, using pruning method during training as well as evaluating algorithm with unseen test dataset we address the overfitting issue. Choosing what fraction of data should be used for training and testing is an open question. We put 30 percent of data for test and 70 percent of data for training.

Different measures are defined for evaluation of performance of prediction models. Two main metrics are precision and recall. In a simple word, in a binary case, precision represent the fact that how many of predicted values as positive is actually positive and recall represent the idea that how many of positive cases are diagnosticated with the algorithm. Choosing between these two measures is dependent on the application. For example if there is a highly contagious disease and we want to put all contaminated persons in a quarantine, we need to focus on recall value, because even if we miss one case, it could distribute the virus, however, there is a trade off between recall and precision. In the mentioned example there should be some person that indicated as contaminated with virus where as they are not. Therefor, the algorithm precision is not high. Table 1 which is called confusion matrix is a method to represent the overall functionality of the algorithm (Branco et al. 2015).

According to Table 1 and the definition, mathematically precision and recall are defined as

$$Precision = \frac{TP}{TP + FP}, \quad Recall = \frac{TP}{TP + FN}, \tag{5}$$

Because of trade of between these two metrics other metrics are defined to generate a metric considering both values. Among them we use $F - measure$ (Rijsbergen1979) which is defined as

$$F_\beta = \frac{(1+\beta)^2.recall.precision}{\beta^2.recall + precision}, \tag{6}$$

In this study we assume $\beta = 1$ to give same importance to precision and recall. Developing a prediction model has a broad discussion. In this study we are not presenting the best model, however, we present an intuition into the method and be able to present a simple model where a user could estimate the target class without using sophisticated methods. We divided data into training and test set. C5.0 package in R has two options to prune the tree. These parameters include: Confidence factor (CF) and the smallest number of samples that must be put in at least two of the splits (minCases). We tune the algorithm for these two parameters based on training with train data and test the results on test data based on F1 score. We also active the winnow option, which activate the internal feature selection algorithm of the C5.0 package. Fig. 11 represent the tuning parameters results. It's obvious that the minCases is dominant factor. In this study we choose $CF = 0.2$.

Fig. 12 represent the variation of F1-score for different clusters (or classes) with respect to the minCases.

A decision tree could grow to very low levels and predict all training data accurately unless there is duplicated data with two different target value. The decision tree algorithm in this study provides very good results even the for test data. However, the scope of this study is to generate a simple relationship to evaluate the simulation results. In the case of going to very sophisticated methods, other algorithms (random forest, conditional random forest, neural networks, ...) could be much more accurate. Consequently, In this study we do not activate the boosting capability of the C5.0 algorithm. According to Fig. 12 lower minCases and higher CF provides a better results. It is worth to not that the results are based on evaluating algorithm on test data, therefore, higher values are not representing overfitting. However, as we mentioned earlier, we are not presenting the best model, rather we are presenting a simplest model that in general represent the feature effects. First we start with a very simple model. A simple model needs a strong pruning process. Therefore we use $minCases = 100, CF = 0.2$. Fig. 13 shows the classification decision tree. Table 2 shows the confusion matrix of applying the prediction model on test dataset and Table 3 represent the precision, recall, and F1-score for each individual class.

Summary of attribute usage in decision tree are according to Table 4

Now we decrease the $minClass$ to 20 to have a more accurate model we call this model M2. Fig. 14 represents the resultant model.

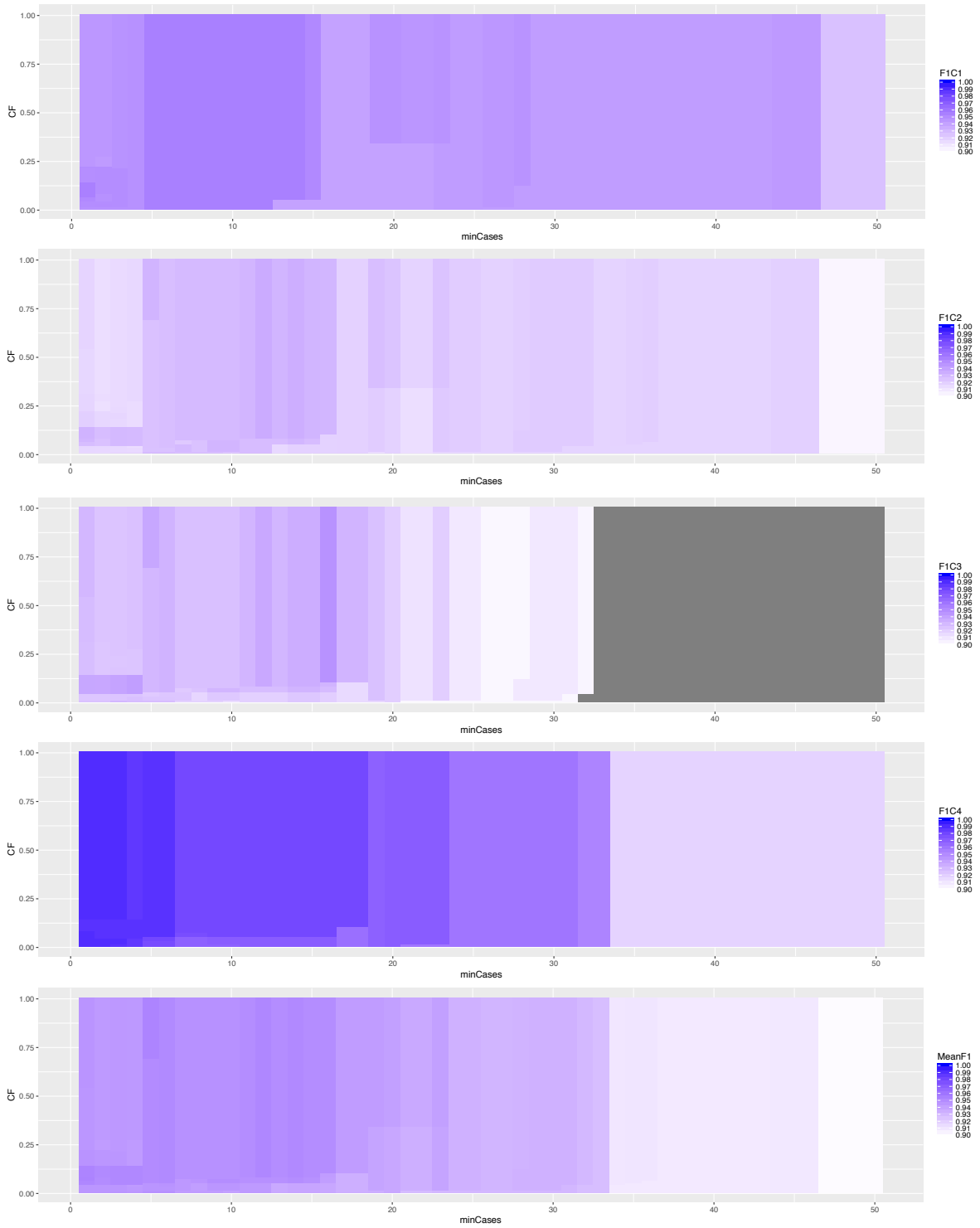

Figure 11. Results of tuning C5.0 algorithm through grid search on confidence factor (CF) and the smallest number of samples that must be put in at least tow of the splits (minCases). Color variation represents F1-score.

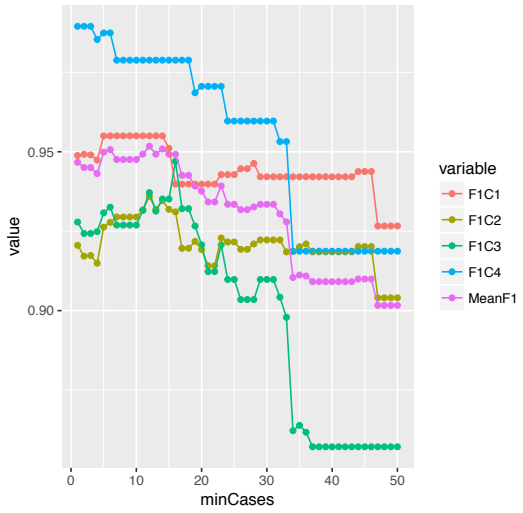

Figure 12. Variation of F1-score for $CF = 0.2$ for different cluster. Mean-F1 score represent the geometrical mean of F1-score of other clusters.

Confusion matrix, model scores, and attribute usage are presented in Table 5, Table 6, and Table 7.

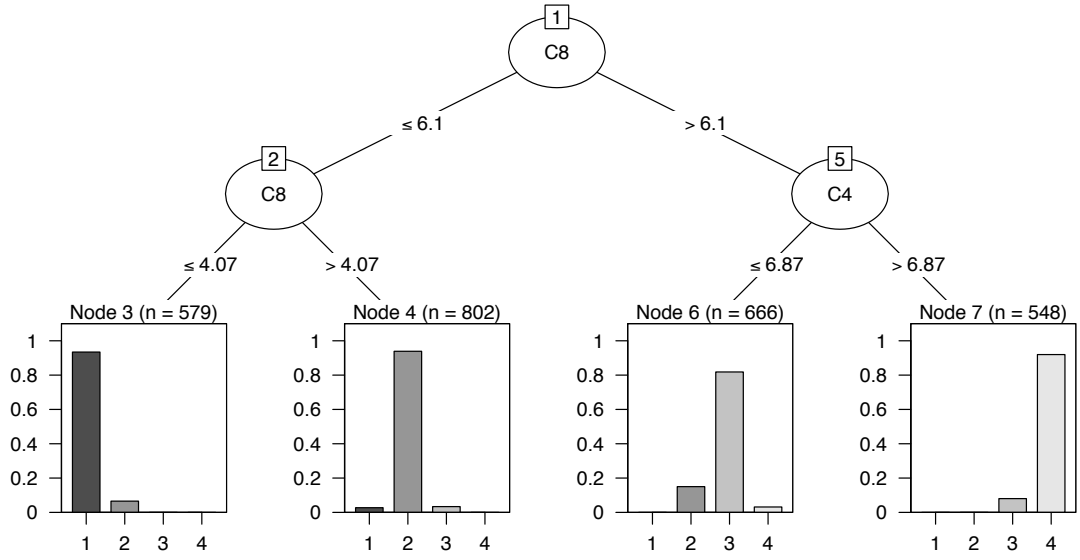

Figure 13. First prediction model (M1) for classifying ground motion simulation process based on goodness of fit scores of two metrics (i.e., C8 (Response Spectra) and C4(Total Energy)). Number above the box represent the number of training data that end up to that specific node. Bar plots represent the probability of each class under the related condition.

Table 2. Confusion Matrix for first prediction model.

		Prediction			
Reference	C	1	2	3	4
	1	240	25	0	0
	2	13	299	21	0
	3	0	38	211	9
	4	0	0	31	226

7 DISCUSSION

According to the results Response Spectra (C8) is the most important metric in classifying the stations. Data percentage in Table. 4 represent the amount of data that the attribute is used in classifying them. After C8, Arias Intensity Value is the most effective metric. With $C8 \leq 4.07$ & $4.07 < C8 \leq 6.1$ and $6.1 < C8$ & $6.87 \leq C4$ and $6.87 \leq C4$ &$6.1 \leq C8$ one can classify the simulation as poor, fair, good, and excellent, respectively, with a high confidence. In this case minimum F1-score occurs for Class 3 and maximum F1-score occurs for Class 1. Pruning process reduce the effect of oversampling in cluster 4. In the second model, which is more accurate from the first model for all metrics, other than Response Spectra and Total Energy, 3 other metrics including: Peak Velocity, Peak Acceleration, and Cross Correlation, also determines the classes. Fig. 15 represents the variation of Response Spectra score in different distances for different velocity models and components. As one can see it is fairly well distributed in different score variation in respect to response. There is differences in median value in different components. The median value improves from CVMH+without GTL layer to CVMS.

Analysis of statistical significance for relevance of results to earthquake, velocity model, frequency band, magnitude, distance of stations, and components needs comprehensive study beyond clustering and classification. However, we illustrate the effects without making strong

Table 3. Precision, Recall, and F1-score of the first prediction model calculated from the confusion matrix.

	Precision	Recall	F1 Score
Class 1	0.9056604	0.9486166	0.9266409
Class 2	0.8978979	0.8259669	0.8604317
Class 3	0.8178295	0.8022814	0.8099808
Class 4	0.8793774	0.9617021	0.9186992

Table 4. Percentage of data that used the attribute to classify them.

Data percentage (%)	Attribute	Metric
100.00	C8	Response Spectra
46.78	C4	Total Energy

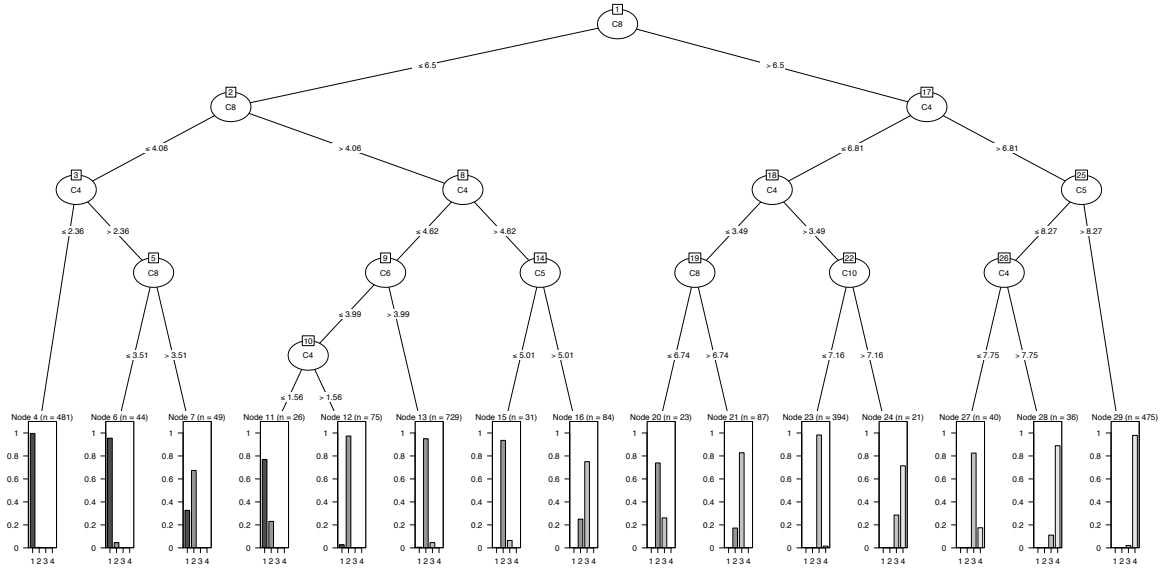

Figure 14. Second classification model

Table 5. Confusion matrix

		Prediction			
Reference	C	1	2	3	4
	1	242	20	0	0
	2	11	330	15	0
	3	0	12	238	4
	4	0	0	10	231

Table 6. Recall and precision

	Precision	Recall	F1 Score
Class 1	0.9236641	0.9565217	0.9398058
Class 2	0.9269663	0.9116022	0.9192201
Class 3	0.9370079	0.9049430	0.9206963
Class 4	0.9585062	0.9829787	0.9705882

Table 7. Data usage percentile

Data percentage (%)	Attribute	Metric
100.00	C8	Response Spectra
100.00	C4	Total Energy
31.87	C6	Peak Velocity
25.66	C5	Peak Acceleration
19.50	C10	Cross Correlation

decision whether they affect the results or not. More comprehensive study is undergoing with Taborda et al. (2016) dataset.

Finally, we represent the application of these models on classifying stations on a geographical representation. Fig. 16 represent the application of the first decision tree algorithm on the simulation of Chino Hills earthquake. In order to keep consistency in color code, and be able to compare the results with other published results (e.g., Taborda & Bielak 2013, 2014; Taborda et al. 2016) we assign 3,5,7, and 9 to cluster 1(poor),2(fair),3(good), and 4(excellent), respectively.

Subsequently, Fig. 17 represent of application of M2 metric on data.

These figures are important from different point of view. As we mentioned before and illustrated in different figures, the GOF results for different component could be different. This can happen for many reason, for example not accurate orientation of station or simply location of station where makes difference for vertical and horizontal incidents. The fact that which component we should use as a final decision is beyond the scope of this paper. However, using different data in clustering and generating classification models should not affect the

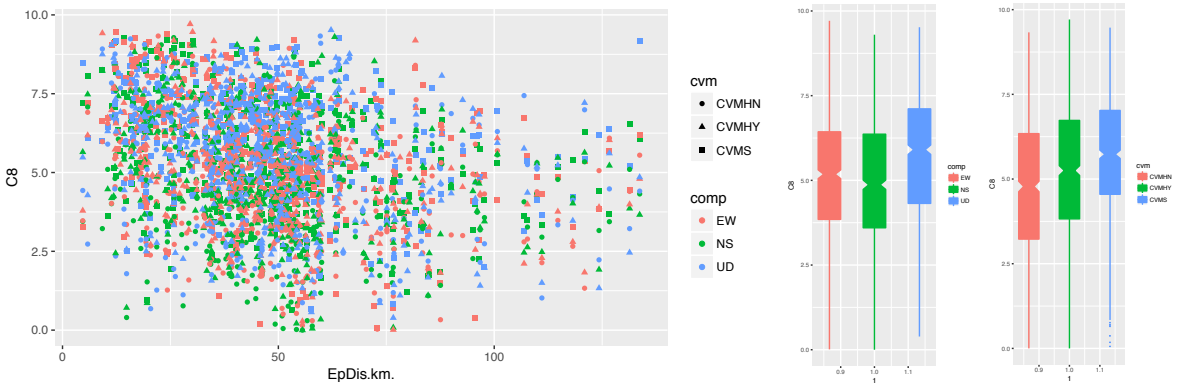

Figure 15. Variation of C8 (Response Spectra) score with distance. Velocity models are represented in different shapes and components in different colors.

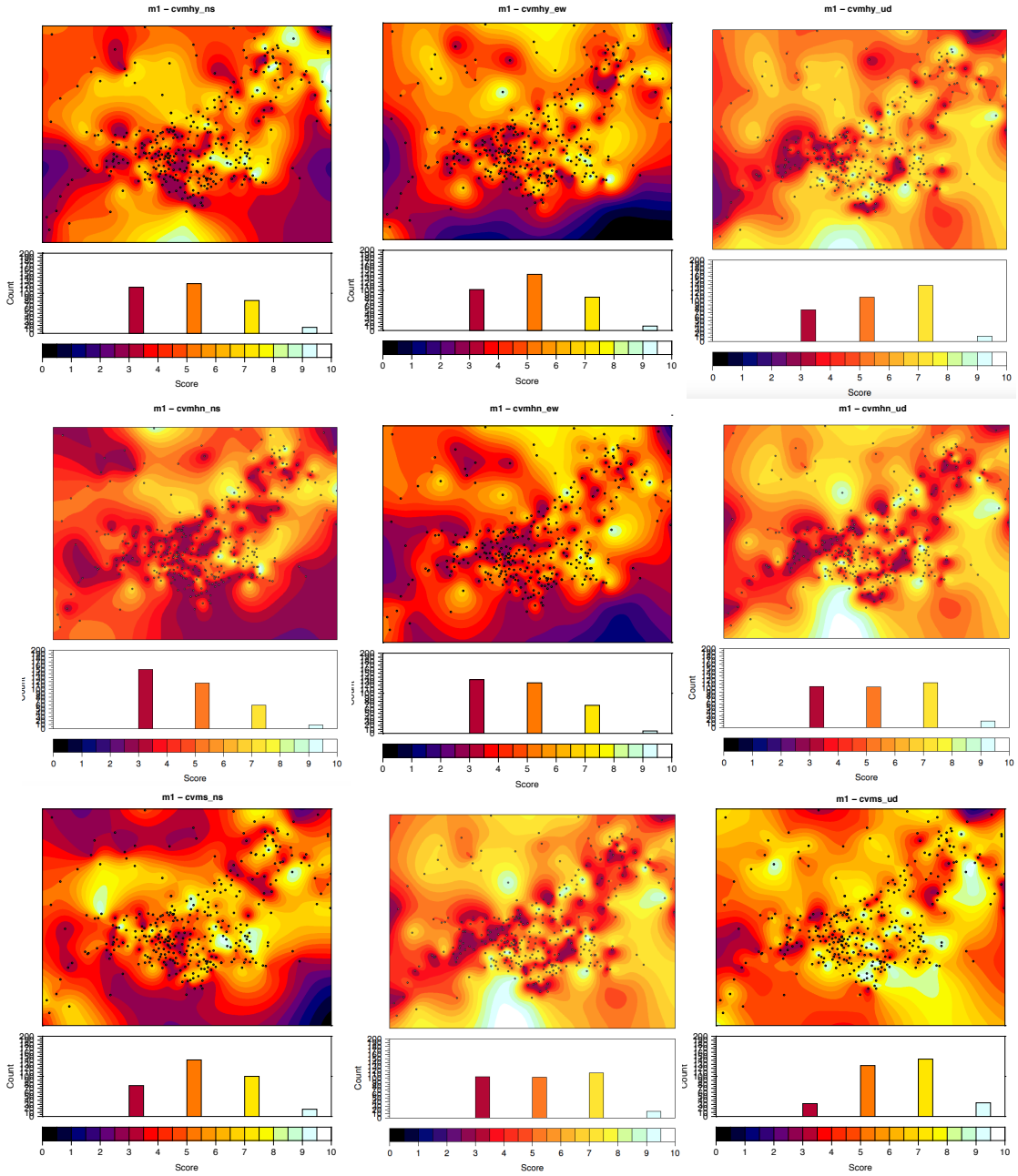

Figure 16. M1 GOF-score for 3 velocity models and 3 components for Chino Hills earthquake simulation (max =4Hz)

results. We consider the GOF of two seismograms (data and synthetic) as an observation to study the relationship between metrics not components. We only distinguished data based on components for presentation purposes. In all steps we use all data. This argument is also correct for velocity model. A velocity model could result in different accuracy and GOF scores. Not surprisingly, the algorithm will classify them in different classes. The model predicts

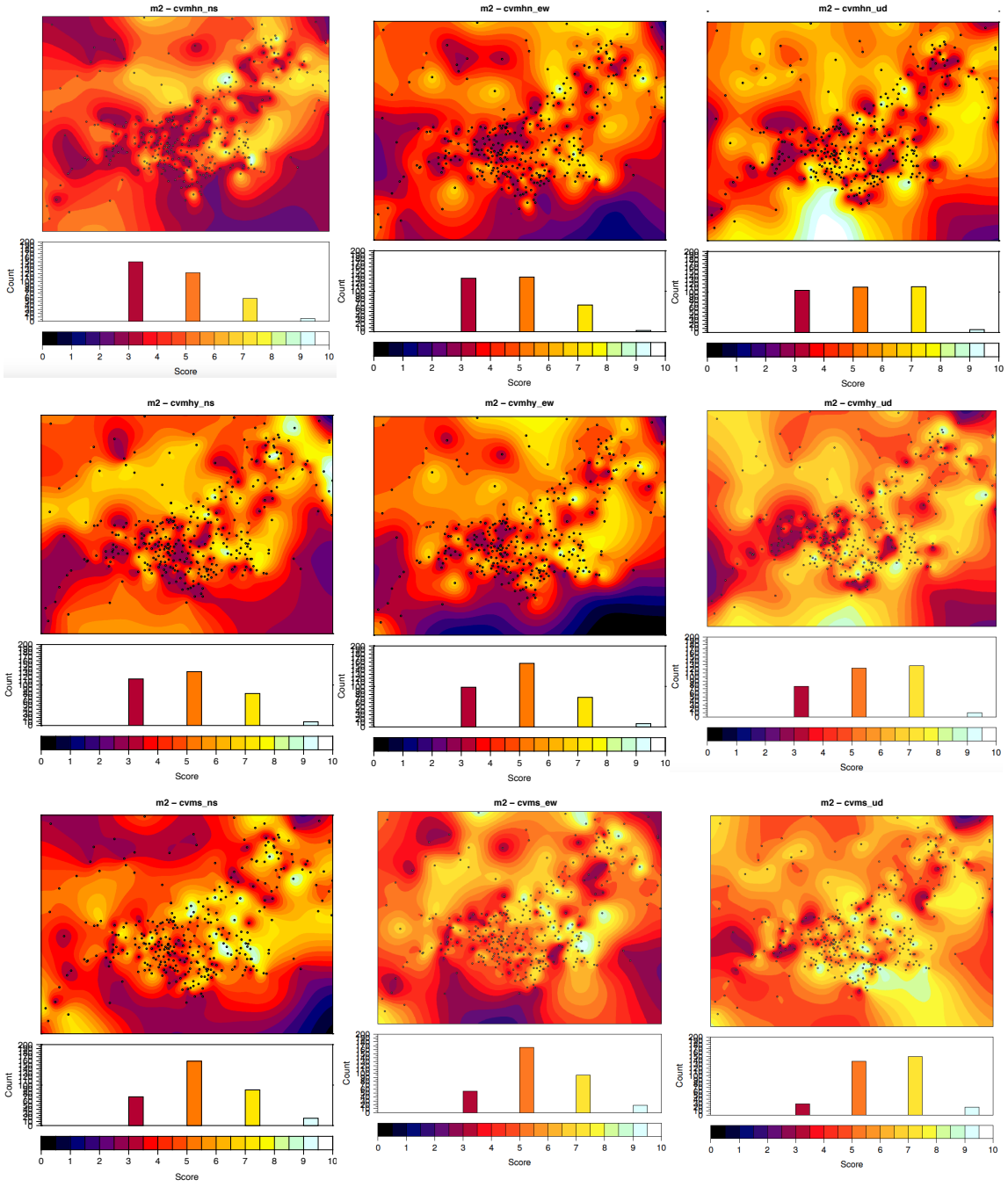

Figure 17. M2 GOF-score for 3 velocity models and 3 components for Chino Hills earthquake simulation (max =4Hz)

different results for the same pair of data and synthetic for different components and velocity models simply because they are different.

8 CONCLUSIONS

In this study we use two machine learning algorithms to address the long time question about choosing the appropriate metrics to evaluate the ground motion simulation process. Assuming hypothetical stations, we labeled the dataset through constraint *k-means* algorithm with subspace analysis. Using the labeled data we develop a decision tree algorithm to evaluate the ground motion simulation. We generate two models where the first model only uses two metrics to evaluate the stations and the second model which is more accurate and also more complicated than the first model uses 5 metrics. According to our analysis, Response Spectra is the most important metric in classifying the stations. After response spectra, Total Energy are mostly used. We present the results for separate velocity models and components, however, a comprehensive study is needed to address the significance of the results or variations to earthquake magnitude, velocity model, component, station distance from epicenter, earthquake depth, and so on. The proposed model could be used with high accuracy in determining the accuracy of simulation uniformly among physics based ground motion simulation developers.

REFERENCES

Anderson, J. G., 2004. Quantitative measure of the goodness-of-fit of synthetic seismograms, in *Proc. 13th World Conf. on Earthquake Eng.*, Int. Assoc. Earthquake Eng., Vancouver, British Columbia, Canada, Paper 243.

Branco, P., Torgo, L., & Ribeiro, R., 2015. A survey of predictive modelling under imbalanced distributions, *arXiv preprint arXiv:1505.01658*.

Duda, R. O., Hart, P. E., & Stork, D. G., 1973. *Pattern classification*, Wiley, New York.

Dy, J. G. & Brodley, C. E., 2004. Feature selection for unsupervised learning, *Journal of machine learning research*, **5**(Aug), 845–889.

Fayyad, U. M., 1996. Data mining and knowledge discovery: Making sense out of data, *IEEE Expert: Intelligent Systems and Their Applications*, **11**(5), 20–25.

Guyon, I. & Elisseeff, A., 2003. An introduction to variable and feature selection, *Journal of machine learning research*, **3**(Mar), 1157–1182.

Hartigan, J. A., 1985. Statistical theory in clustering, *Journal of classification*, **2**(1), 63–76.

Hornik, K., Buchta, C., & Zeileis, A., 2009. Open-source machine learning: R meets Weka, *Computational Statistics*, **24**(2), 225–232.

Jain, A. K. & Dubes, R. C., 1988. *Algorithms for clustering data*, Prentice-Hall, Inc.

Jain, A. K., Murty, M. N., & Flynn, P. J., 1999. Data clustering: A review, *ACM Comput. Surv.*, **31**(3), 264–323.

Khoshnevis, N. & Taborda, R., 2015. Sensitivity of ground motion simulation validation criteria to filtering, International Conference on Applications of Statistics and Probability in Civil Engineering (ICASP) (12th: 2015), This collection contains the proceedings of ICASP12, the 12th International Conference on Applications of Statistics and Probability in Civil Engineering held in Vancouver, Canada on July 12-15, 2015. Abstracts were peer-reviewed and authors of accepted abstracts were invited to submit full papers. Also full papers were peer reviewed. The editor for this collection is Professor Terje Haukaas, Department of Civil Engineering, UBC Vancouver.

Kristeková, M., Kristek, J., Moczo, P., & Day, S. M., 2006. Misfit criteria for quantitative comparison of seismograms, *Bull. Seismol. Soc. Am.*, **96**(5), 1836–1850.

Kristeková, M., Kristek, J., & Moczo, P., 2009. Time-frequency misfit and goodness-of-fit criteria for quantitative comparison of time signals, *Geophys. J. Int.*, **178**(2), 813–825.

Kuhn, M., Weston, S., Coulter, N., & code for C5.0 by R. Quinlan, M. C. C., 2015. *C50: C5.0 Decision Trees and Rule-Based Models*, R package version 0.1.0-24.

MacQueen, J. et al., 1967. Some methods for classification and analysis of multivariate observations, in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, pp. 281–297, Oakland, CA, USA.

Mitchell, T., 1997. *Machine Learning*, McGraw-Hill International Editions, McGraw-Hill.

Olsen, K. B. & Mayhew, J. E., 2010. Goodness-of-fit criteria for broadband synthetic seismograms, with application to the 2008 M_{w} 5.4 Chino Hills, California, earthquake, *Seismol. Res. Lett.*, **81**(5), 715–723.

Parsons, L., Haque, E., & Liu, H., 2004. Subspace clustering for high dimensional data: a review, *Acm Sigkdd Explorations Newsletter*, **6**(1), 90–105.

Quinlan, J. R., 1986. Induction of decision trees, *Machine learning*, **1**(1), 81–106.

Quinlan, J. R., 1993. *C4. 5: programs for machine learning*, Morgan Kaufmann, Burlington, Massachusetts.

Taborda, R. & Bielak, J., 2013. Ground-motion simulation and validation of the 2008 Chino Hills, California, earthquake, *Bull. Seismol. Soc. Am.*, **103**(1), 131–156.

Taborda, R. & Bielak, J., 2014. Ground-motion simulation and validation of the 2008 Chino Hills, California, earthquake using different velocity models, *Bull. Seismol. Soc. Am.*, **104**(4), 1876–1898.

Taborda, R., Azizzadeh-Roodpish, S., Khoshnevis, N., & Cheng, K., 2016. Evaluation of the southern california seismic velocity models through simulation of recorded events, *Geophysical Journal International*, **205**(3), 1342.

Wagstaff, K., Cardie, C., Rogers, S., Schrödl, S., et al., 2001. Constrained k-means clustering with background knowledge, in *ICML*, vol. 1, pp. 577–584.

Watanabe, S., 1985. *Pattern recognition: human and mechanical*, John Wiley & Sons, Inc.

Wei, T. & Simko, V., 2016. *corrplot: Visualization of a Correlation Matrix*, R package version 0.77.

Weiss, G. M. & Provost, F., 2003. Learning when training data are costly: The effect of class distribution on tree induction, *Journal of Artificial Intelligence Research*, **19**, 315–354.